Remarks

Claims 1-7, 10, 13, 15 and 17-28 are currently pending in this application. Claim 17 is currently allowed and Claim 21 is currently objected to as being dependent on a rejected claim. Claim 28 is new. The Applicant respectfully requests reconsideration of the present application in view of the remarks set forth below.

In paragraph 7 of the Final Office Action the Examiner states that the reference referred to as LARS JACOBSEN was first disclosed as a pre-print in November 1999 and, thus, the 37 CFR 1.131 affidavit filed in response to the previous Office Action is ineffective to overcome this reference. The Examiner further states that the Examiner previously provided this information to the Applicant. The Applicant traverses this statement and respectfully points out that while the Notice of References Cited included in the previous Office Action included the November 1999 date, the copy of the reference provided with the previous Office Action did not include evidence of the November 1999 date.

In response to receiving a document without evidence of a publication date, the Applicant's representative searched for the reference on the Internet using the authors' names and the reference title. It was found that the reference was published by the Association for Computing Machinery (ACM) in October of 2001. The Applicant, therefore responded to the previous Office Action with an affidavit filed under 37 CFR 1.131 in order to swear behind the October 2001 date.

In the Final Office Action of 2/15/2007, the Examiner has provided new information regarding JACOBSEN. Specifically, the copy of the reference provided with

the Final Office Action now includes a cover sheet bearing the University of Southern Denmark logo and the November 1999 date. Significantly, the copy provided with the previous Office Action did not include this coversheet and, thus, did not evidence the November 1999 date. Note that these differences between the two copies of the cited reference are reflected in the electronic versions available through PAIR.

The publication date shown on the cover sheet and relied on by the Examiner in the Final Office Action of 2/15/2007 is, thus, new information that was not previously provided to the Applicant. The Applicant traverses the finality of the current office action on the grounds that this new information represents new grounds of rejection. Specifically, the Applicant reasonably relied on the ACM publication date of October 2001 as the initial publication of the reference. Because the Examiner withheld evidence as to the publication date of the cited reference in the previous office action, the Applicant has lost an opportunity to respond to the rejections on their merits. The Applicant, therefore, requests that the Examiner remove the finality of the current office action.

Claim Rejections – 35 U.S.C. § 103

Claims 1-3, 5-7, and 13 are rejected under 35 U.S.C. 103(a) as being unpatentable over "Complexity of Layered Binary Search Trees with Relaxed Balance" by LARS JACOBSEN et al.

Regarding Claim 1,

Claim 1 recites:

1. (Previously Presented) A method of reducing the number of times a tree data structure is rebalanced comprising the steps of:
(a) allowing a sub-tree of the tree data structure to grow until a number of unbalanced levels reaches a threshold greater than one; and
(b) rebalancing the tree data structure when the threshold is reached.

In rejecting Claim 1, the Examiner states "JACOBSON teaches a method of reducing the number of times a tree data structure is rebalanced comprising the steps of: allowing a sub-tree of the tree data structure to grow until a number of unbalance levels reaches a threshold and rebalancing the tree data structure when the threshold is reached." In support of this statement, the Examiner cites teachings of JACOBSON beginning at pg. 2, 2nd paragraph through the last line of pg. 2. It is the Applicant's understanding that the Examiner is suggesting that the teaching of JACOBSON "to monitor path length and rebalance when some limit is exceeded," teaches the claim limitations "*allowing a sub-tree of the tree data structure to grow until a number of unbalanced levels reaches a threshold greater than one.*" The Applicant traverses this suggestion.

It is the position of the Applicant that the cited art merely teaches a non-specific concept of monitoring path length and does not teach allowing growth "*until a number of unbalanced levels reaches a threshold greater than one.*" The cited art must either explicitly or inherently teach all of the limitations of a rejected claim in order for the Examiner to establish a prima facie case for rejection under 103.

It is the position of the Applicant that the cited art does not explicitly teach the above limitations in that monitoring "a path length" is not the same as allowing growth "*until a number of unbalanced levels reaches a threshold greater than one.*" One of

ordinary skill in the art would understand "path length" to mean a number of levels between nodes in the tree data structure. For example, the path length monitored in JACOBSON may be a number of levels between the root node and a leaf node. The "path lengths" of JACOBSON are different from the "number of unbalances levels" recited in Claim 1 because the "path lengths" may include both balanced and unbalanced levels, and not merely unbalanced levels. The Applicant is unable to identify any teaching within JACOBSON that the taught "path length" includes any distinction between balanced and unbalanced levels. Thus, JACOBSON's monitoring of "path length" does not provide information regarding "*a number of unbalanced levels*" and, thus, cannot teach halting growth when this number "*reaches a threshold greater than one.*" JACOBSON, therefore, does not teach allowing growth "*until a number of unbalanced levels reaches a threshold greater than one,*" as recited in Claim 1.

With regard to inherency, MPEP §2112 provides that "[t]he fact that a certain result or characteristic may occur or be present in the prior art is not sufficient to establish the inherency of that result or characteristic" citing In re Rijckaert, 9 F.3d 1531, 1534, 28 USPQ2d 1955, 1957 (Fed. Cir. 1993). Further, "[i]n relying upon the theory of inherency, the examiner must provide a basis in fact and/or technical reasoning to reasonably support the determination that the allegedly inherent characteristic necessarily flows from the teachings of the applied prior art" citing Ex parte Levy, 17 USPQ2d 1461, 1464 (Bd. Pat. App. & Inter. 1990).

In the teachings cited by the Examiner, the limitations of Claim 1 are not inherent because they do not "necessarily flow" from the teachings. For example, as discussed above, the "path lengths" monitored in JACOBSON may be lengths of entire paths rather

than merely an unbalanced portion of these paths. Thus, JACOBSON does not necessarily include "*allowing a sub-tree of the tree data structure to grow until a number of unbalanced levels reaches a threshold greater than one*," and the limitations of Claim 1 do not "necessarily flow" from the cited art and JACOBSON, therefore, does not inherently teach the limitations of Claim 1.

For at least these reasons, the Applicant believes that Claim 1 and those claims that depend therefrom are allowable.

Regarding Claim 2:

Claim 2 recites:

2. (Previously Presented) The method of claim 1 wherein the threshold is log_2n for a tree data structure having about n nodes.

In rejecting Claim 2, the Examiner states "Official Notice is taken in that functions, such as random and logarithmic, are used to determine a number and would be obvious to one of ordinary skill in the art that this number [log_2n] is the threshold used for starting the rebalancing." The Examiner's statement includes two suggestions of fact: first, "that functions, such as random and logarithmic, are used to determine a number," and second, that it "would be obvious to one of ordinary skill in the art that this number [log_2n] is the threshold used for starting the rebalancing." The Applicant specifically traverses the second suggestion of fact.

As specified in MPEP §2144.03(A), "[o]fficial notice unsupported by documentary evidence should only be taken by the examiner where the facts asserted to be well-known, or to be common knowledge in the art are capable of instant and unquestionable demonstration as being well-known." The Applicant is unaware of any common knowledge or facts well known in the art that would support the Examiner's

second suggestion of fact. The Applicant, therefore, specifically requests that the Examiner provide documentary evidence, as required by MPEP §2144.03(C), in the next action to support the Examiner's suggestion that it "would be obvious to one of ordinary skill in the art that this number [$\log_2 n$] is the threshold used for starting the rebalancing," or withdraw the rejection of Claim 2.

The Applicant believes that Claim 2 is further allowable for at least the same reasons as Claim 1, from which it depends.

Regarding Claim 3:

Claim 3 recites:

> *3. (Previously Presented) The method of claim 1 wherein the threshold is a constant number of levels greater than a level of a balanced portion of the tree data structure.*

The Examiner does not appear to specifically address the limitations of Claim 3 in the current Office Action. Specifically, nowhere does the Examiner state that the claim limitations of "*the threshold is a constant number of levels greater than a level of a balanced portion of the tree data structure*" are taught by the cited reference. The Applicant is unable to identify teachings of these limitations within the cited art. For example, the monitoring of "path length" as taught by JACOBSON is distinguished from the limitations of Claim 3 in that the threshold used is a "*a constant number of levels greater than a level of a balanced portion of the tree data structure.*" Because, as discussed above, the monitoring of JACOBSON is not taught to distinguish between balanced and unbalanced levels, the Applicant does not see how JACOBSON could teach these limitations, which do require a distinction between balanced and unbalanced levels.

The Examiner is respectfully reminded that the cited art must include all of the limitations of the claim in order to establish a prima facie case for rejection under 103. The Applicant, therefore, requests that the Examiner specifically cite teachings of "*the threshold is a constant number of levels greater than a level of a balanced portion of the tree data structure*," or allow Claim 3.

The Applicant believes that Claim 3 is further allowable for at least the same reasons as Claim 1, from which it depends.

Regarding Claim 5,

Claim 5 recites:

> *5. (Previously Presented) A method of deferring the rebalancing of a tree data structure comprising the steps of:*
> *(a) allowing a sub-tree of the tree data structure to grow to an unbalanced length greater than one; and*
> *(b) rebalancing the tree data structure when the unbalanced length of the sub-tree reaches a threshold level.*

In rejecting Claim 5, the Examiner cites the same art as cited in the rejection of Claim 1. However, Claim 5 includes limitations specifying that the "*unbalanced length*" is that of a "*sub-tree*." The Applicant is unable to find any teaching of these limitations in the cited art. Specifically, as discussed above, these limitations are not explicitly taught because the monitored "path lengths" (JACOBSON page 2) are not the same as "*unbalanced*" path lengths. Likewise, it is the position of the Applicant that the "path lengths" of JACOBSON are not equilivant to "*length of the sub-tree*" as recited in Claim 5. As discussed above, the "path length" of JACOBSON may be the entire length from a root node to a leaf node. In contrast, a length of a sub-tree may include merely a fraction of this path length, e.g. the length from an intermediate node to a leaf node. The "path

length" of JACOBSON is, thus, distinguished from "*the unbalanced length of the sub-tree,*" as recited in Claim 5.

Further, it is the position of the Applicant that these limitation are not inherently taught because such a teaching does not "necessarily flow" from the teachings of JACOBSON. For example, the monitored path lengths of JACOBSON may include a balanced length of an entire tree data structure, rather than "*the unbalanced length of the sub-tree.*" The limitations "*the unbalanced length of the sub-tree,*" therefore, do not "necessarily flow" from the teachings of JACOBSON, and these limitations are not inherently taught by the cited art.

The Applicant, therefore, requests that the Examiner specifically point out these teachings within the cited art or allow Claim 5 and those claims that depend therefrom.

Regarding Claim 6,

Claim 6 recites:

6. (Original) The method of claim 5 wherein the threshold level is $\log_2 n$ for a tree data structure having about n nodes.

The Applicant believes that Claim 6 is allowable for at least the reasons discussed above with respect to Claim 2, and Claim 5 from which it depends.

Regarding Claim 7,

Claim 7 recites:

7. (Original) The method of claim 5 wherein the threshold level is a constant number of levels greater than a level of a balanced portion of the tree data structure.

The Applicant believes that Claim 7 is allowable for at least the reasons discussed herein with respect to Claim 3, and Claim 5 from which it depends.

Regarding Claim 13,

Claim 13 recites:

13. (Previously Presented) A system for deferring the rebalancing of a tree data structure comprising:

(a) a memory for storing the tree data structure; and

(b) a processor coupled to the memory, the processor operable to track the performance of operations upon the tree data structure and rebalance the tree data structure when a number of unbalanced levels within a sub-tree of the tree data structure reaches a threshold greater than one.

The Applicant believes that Claim 13 is allowable for at least the reasons discussed herein with respect to Claims 1 and 5.

Claims 4, 10, 15 18-20 and 22-27 are rejected under 35 U.S. C. 103(a) as being unpatentable over "Complexity of layered Binary Search Trees with Relaxed Balance" by LARS JACOBSEN et al. in view of "Relaxed AVL Trees, Main-Memory Databases, and Concurrency" by OTTO NURMI ET AL.

Regarding Claim 4,

The Applicant believes that Claim 4 is allowable for at least the same reasons as Claim 1, from which it depends.

Regarding Claim 10,

The Applicant believes that Claim 10 is allowable for at least the same reasons as Claims 1 and 5.

Regarding Claim 15,

Claim 15 recites:

15. (Previously Presented) A system comprising:

means for storing a tree data structure;

means for tracking the execution of operations upon the tree data structure; and

means for rebalancing the tree data structure when an unbalanced sub-tree of the tree data structure reaches a threshold level greater than one, the rebalancing including a first rebalancing phase in which rebalancing

operations are executed in parallel and nodes of the unbalanced sub-tree are unlocked, and a second rebalancing phase in which different rebalancing operations are executed.

In rejecting Claim 15, the Examiner cites JACOBSON and in addition states:

> NURMI teaches the rebalancing operations including a first rebalancing phase in which rebalancing operations are executed in parallel and nodes of the unbalanced sub-tree are unlocked (via decreasing the tag-value during the rebalancing process that has found an unbalanced node based on the tag-value), and a second rebalancing phase in which different rebalancing operations are executed (via rotating the tree to put the tree in balance)."

The Applicant traverses this statement. Specifically the Applicant is unable to identify any teaching within the cited art of "*a first rebalancing phase in which rebalancing operations are executed in parallel and nodes of the unbalanced sub-tree are unlocked.*"

While NURMI does teach "tasks are, for simplicity described without locking," NURMI goes on to state "[i]n Section 6, we specify a simple locking method that enables updaters and rebalancers to operate concurrently in a relaxed AVL tree," (pg. 3, 1st paragraph). Further, on pg. 18 (Section 6) NURMI discusses several different locking mechanisms (r-locks, w-locks and e-locks) that are required for "an arbitrary number of processes to operate concurrently."

It is the Applicant's position that a person of ordinary skill in the art would interpret these teachings as follows: a) NURMI teaches a simplistic system that can be performed without locking but that does not allow concurrent operations, and b) a more practical system that allows concurrent processes but requires locking. For example, the teaching of "a simple locking method that enables ... to operate concurrently" suggests that the locking method is required for concurrent operation. Likewise, in Section 6, the

locking mechanisms are suggested as being required for "an arbitrary number of processes to operate concurrently." Therefore, NURMI does not teach a system in which concurrent processes operate on unlocked nodes.

It is the Applicant's position that these teachings, therefore, do not include the limitations of Claim 15. Specifically, Claim 15 recites, "*rebalancing operations are executed in parallel*," while "*nodes of the unbalanced sub-tree are unlocked*," which is not taught by the cited art. The Applicant, therefore, requests that the Examiner specifically point out a teaching within NURMI that includes *parallel rebalancing operations* on *nodes of the unbalanced sub-tree that are unlocked*, or allow Claim 15.

The Applicant further believes that Claim 15 is allowable for the same reasons as Claims 1 and 5.

Regarding Claim 18,

Claim 18 recites:

> *18. (Original) A method of deferring the rebalancing of a tree data structure comprising the steps of:*
> *(a) tracking the performance of operations upon the tree data structure; and*
> *(b) rebalancing the tree data structure when an unbalanced sub-tree of the tree data structure reaches a threshold level greater than one, the rebalancing further comprising executing simultaneous rebalancing operations on the tree data structure including performing any first phase operation task of each of the simultaneous rebalancing operations in a first phase using parallel processes, developing a set of serial rebalancing operations during the first phase, and performing any second phase operation task of each of the simultaneous rebalancing operations in a second phase, the second phase operation task having at least one of the set of serial rebalancing operations.*

The Applicant believes that Claim 18 is allowable for at least the same reasons as Claims 1 and 5. Specifically, JACOBSON does not teach "*when an unbalanced sub-tree of the tree data structure reaches a threshold level greater than one.*"

Regarding Claim 19,

Claim 19 recites:

19. (Previously Presented) A method of rebalancing a tree data structure, the method comprising:
allowing a sub-tree of the tree data structure to grow unbalanced until a threshold level is reached;
developing a first set of rebalancing operation tasks, the first set of operation tasks operable in parallel on one or more unlocked nodes of the tree data structure during a first phase of the rebalancing;
developing a second set of rebalancing operation tasks during execution of the first set of rebalancing operation tasks; and
executing the second set of rebalancing operation tasks during a second phase of the rebalancing.

The Applicant believes that Claim 19 is allowable for at least the same reasons as Claim 15. Specifically, JACOBSON does not teach "*the first set of operation tasks operable in parallel on one or more unlocked nodes.*"

Regarding Claim 20,

Claim 20 recites:

20. (Previously Presented) The method of claim 19, wherein execution of the second set of rebalancing operation tasks is performed without navigating between nodes of the sub-tree.

In rejecting Claim 20, the Examiner states "[i]t is inherent to the teachings of NURMI that the rotation operations executing in the second phase do not navigate to the nodes, because they were already traversed in the first phase in determining whether an unbalance exists." The Applicant traverses this statement on several grounds.

First, the Examiner is mischaracterizing the claim language. Claim 20 does not recite "not navigating to nodes" as suggested by the Examiner. Rather Claim 20 recites "*without navigating between nodes.*"

Second, the Examiner does not appear to have properly established that "without *navigating between nodes*" is inherent to the teachings of NURMI. As discussed above, for a teaching to be inherent it must "necessarily flow" from teachings of the art.

NURMI teaches a system including a Phase 1 and a Phase 2. In Phase 1, tags are changed by an updater to non-zero values on nodes involved in an insertion or deletion (pg. 7, 1st full paragraph). The updater may be involved in updating many different nodes. These tags allow unbalanced nodes to be identified by a rebalancer in Phase 2. See, for example, page 10, 2nd full paragraph which states "Assume that the rebalancer has found a node p with tag(p) = 0..." In Phase 2, each node is revisited and those nodes including changed tags are rebalanced if necessary. This rebalancing involves making changes to both the node with the non-zero tag and its parent node, (page 10, 4th full paragraph).

The Examiner's reasoning as to why limitations of Claim 20 are inherent to the teachings of NURMI consists of "because they [the nodes] were already traversed in the first phase." The Applicant respectfully points out that if this were true, there would be no need to have tags in the first place because there would be no need to re-identify nodes previously involved in insertion or deletion.

The Examiner's reasoning appears to be contrary to the teachings of NURMI in further ways. For example, the updater would be expected to make further insertions or deletions at other nodes during Phase 1 and, therefore, may no longer be at a first node that requires rebalancing in the second phase. In another example, operations in the different phases of NURMI are performed by different processes, e.g. the updater and the rebalancer. Thus, for the sake of argument, even if the updater were to stay at a node, the

node would still have to be found by the rebalancer. For at least these reasons, the claim limitations, "*without navigating between nodes of the sub-tree*" does not necessarily flow from the cited teachings, and these limitations are not inherent to the teachings of NURMI.

Third, as discussed above, Phase 2 of NURMI involves changes to more than one node and NURMI does not teach how both of these nodes may be changed without navigating between at least these two nodes. As discussed on page 12 of NURMI, Phase 2 includes performing a "rotation depicted in Fig. 6." This rotation includes performing operations on nodes c and p to produce nodes c' and p'. For example, the production of nodes c' and p' appear to involve changing pointers at both nodes c and p. The Applicant points out that NURMI does not teach how these operations could be performed without at least navigating between nodes c and p. Thus, it appears that NURMI teaches that there is navigation between nodes in Phase 2, and the teachings of NURMI are not performed "*without navigating between nodes*." The Applicant, therefore, requests that the Examiner specifically explain how the operations illustrated in Fig. 6 of NURMI are taught to be performed "*without navigating between nodes of the sub-tree*," or allow Claim 20.

For at least these reasons, and those discussed above with respect to Claims 1, 5 and 19, the Applicant believes that Claim 20 is allowable.

Regarding Claim 22,

Claim 22 recites:

22. (Previously Presented) A method of rebalancing a tree data structure, the method comprising:
allowing a sub-tree of the tree data structure to grow unbalanced until a threshold level is reached;

executing a first set of rebalancing operation tasks during a first rebalancing phase, the first rebalancing phase being characterized by navigation between nodes of the sub-tree; and

executing a second set of rebalancing operation tasks during a second rebalancing phase, the second rebalancing phase including navigation to two or more nodes of the sub-tree, the navigation being independent of pointers between nodes of the sub-tree.

In rejecting Claim 22 the Examiner states "NURMI teaches ... the second rebalancing phase including navigating to two or more nodes of the sub-tree, the navigation being independent of pointers between nodes of the sub-tree (via rotating the tree to put the tree in balance)." The Applicant traverses this statement.

As discussed above with regard to Claim 20, NURMI does not teach how the rotation taught in Fig. 6 of NURMI could be accomplished without navigation between nodes. Further, the Applicant is unable to identify any teaching within NURMI of how this navigation could be performed without the use of pointers. The Applicant, therefore, respectfully requests that the Examiner specifically explain how NURMI teaches navigating between the nodes of Fig. 6 without the use of pointers between these nodes, or allow Claim 22 and those claims that depend therefrom.

Regarding Claims 23 and 24,

The Applicant believes that Claims 23 and 24 are allowable for at least the same reasons as Claim 22 from which they depend.

Regarding Claim 25,

Claim 25 recites:

25. (Previously Presented) A method of maintaining a tree data structure, the method comprising:

allowing the tree data structure to grow unbalanced;

performing a first set of rebalancing operation tasks during a first rebalancing phase on a plurality of nodes in the tree data structure, the first set of rebalancing operation tasks being configured for execution while the

plurality of nodes are unlocked and for insertion and deletion of nodes; and

performing a second set of rebalancing operation tasks on the plurality of nodes in a second rebalancing phase, the second set of rebalancing operation tasks being different than the first set of rebalancing operation tasks and being configured for further operations on the plurality of nodes, the second rebalancing phase occurring after completion of the first rebalancing phase.

The Applicant believes that Claim 25 is allowable for at least the same reasons as Claim 15. Specifically, NURMI does not teach a "*first set of rebalancing operation tasks being configured for execution while the plurality of nodes are unlocked.*"

Regarding Claim 26,

Claim 26 recites:

26. (Previously Presented) The method of claim 25, wherein the first set of rebalancing operation tasks are performed in parallel.

The Applicant believes that Claim 26 is allowable for at least the same reasons as Claim 15, and Claim 25 from which it depends. Specifically, NURMI does not teach a "*first set of rebalancing operation tasks being configured for execution while the plurality of nodes are unlocked*" in combination with "*the first set of rebalancing operation tasks are performed in parallel.*"

Regarding Claim 27,

Claim 26 recites:

27. (Previously Presented) A method comprising:

storing a tree data structure;

tracking the execution of operations upon the tree data structure; and

rebalancing the tree data structure when an unbalanced sub-tree of the tree data structure reaches a threshold level greater than one, the rebalancing including a first rebalancing phase in which rebalancing operations are executed in parallel and nodes of the unbalanced sub-tree are unlocked, and a second rebalancing phase in which different rebalancing operations are executed.

The Applicant believes that Claim 26 is allowable for at least the same reasons as Claims 1, 5 and 15. Specifically, the cited art does not teach "*when an unbalanced sub-tree of the tree data structure reaches a threshold level greater than one*," or "*rebalancing operations are executed in parallel and nodes of the unbalanced sub-tree are unlocked.*"

Regarding New Claim 28,

Claim 28 recites:

28. (New) The method of Claim 10, wherein the first and second set of element state transitions each include changing pointers to nodes of the tree data structure.

The Applicant believes that Claim 28 is allowable for at least the reason that the cited art does not include a first set of element state transitions that includes changing pointers. Specifically, those operations that the Examiner suggests teach the first set of element state transitions include changing the value of a tag and not changing a pointer.

Applicant believes that all pending claims are allowable and respectfully requests that the Examiner issue a Notice of Allowance. Should the Examiner have questions, the Applicant's undersigned representative may be reached at the number provided.

Respectfully submitted,

Clifford L. Hersh

Date: April 16, 2007

Steven M. Colby, Ph.D. Reg. No. 50,250

Carr & Ferrell *LLP*
2200 Geng Rd.
Palo Alto, CA 94303
Phone (650) 812-3424
Fax (650) 812-3444